UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/606,641	06/29/2000	Kimberly J. Rush	06576-105006	7703

7590   03/26/2004

King & Spalding
191 Peachtree Street N E 45th Floor
Atlanta, GA 30303

EXAMINER
CHOUDHURY, AZIZUL Q

ART UNIT	PAPER NUMBER
2143	

DATE MAILED: 03/26/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

PTO-90C (Rev. 10/03)

	Application No.	Applicant(s)
Office Action Summary	09/606,641	RUSH ET AL.
	Examiner	Art Unit
	Azizul Choudhury	2143

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM
THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed
  after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any
  earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on <u>*15 December 2003*</u>.
2a) ☒ This action is **FINAL.**   2b) ☐ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is
closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) <u>*21-40*</u> is/are pending in the application.
   4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) <u>*21-40*</u> is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.
10) ☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.
    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
    a)☐ All   b)☐ Some * c)☐ None of:
    1. ☐ Certified copies of the priority documents have been received.
    2. ☐ Certified copies of the priority documents have been received in Application No. _____.
    3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage
       application from the International Bureau (PCT Rule 17.2(a)).
    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
   Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413)
   Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Detailed Action

This action is in response to the Amendment received on December 15, 2003.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that

form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 21-25 and 30-40 are rejected under 35 U.S.C. 102(b) as being

anticipated by Doyle et al (US Pat No: US005838906A), hereafter referred to as Doyle.

1.   With regards to claim 21, Doyle teaches a method for accessing multiple types of

electronic content, comprising: receiving a request for a computer program to process

an input to obtain an output comprising a type of content that is unknown to the

computer program; selecting at least one segment of computer code from a plurality of

segments of computer code that will enable the computer program to process the input,

when the at least one segment of computer code is executed along with the computer

program, to provide the output comprising the type of content that is unknown to the

computer program; and executing the at least one segment of computer code along with

the computer program to process the input and obtain the output comprising the type of

content that is unknown to the computer program, wherein the plurality of segments of

computer code and the at least one segment of computer code are not executable as

an independent computer program (The examiner concludes after consulting with other

members of the office that the phrase "segment of computer code" is equivalent to programs, program objects and applications. Programs and applications are segments of computer code. The terms application and program may be used interchangeably in the disclosure due to their equivalency to the phrase "segment of computer code." In addition, Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed).

2.   With regards to claim 22, Doyle teaches a method wherein selecting at least one segment of computer code comprises selecting at least two segments of computer code from the plurality of segments of computer code whose combined functionality will enable the computer program to process the input, when the at least two segments of computer code are executed along with the computer program, to provide the output comprising the type of content that is unknown to the computer program (Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together, hence two or more may be selected as claimed).

3.   With regards to claim 23, Doyle teaches a method further comprising configuring the at least two segments of computer code to be executed along; with the computer program in a particular order to provide a desired processing of the input (Doyle teaches

a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together; hence two or more may be selected as claimed. Whenever multiple applications run together to produce a single output, it is inherent that they will perform in order as claimed).

4.   With regards to claim 24, Doyle teaches a method comprising configuring the at least two segments of computer code into a master-slave relationship that causes the execution of one of the at least two segments of computer code to be dependent on the execution of another of the at least two segments of computer code (Doyle discloses how applications can be started as child processes (column 15, line 22, Doyle). When a child process exists, a parent process must exist. This parent-child relationship is equivalent to a master-slave relationship).

5.   With regards to claim 25, Doyle teaches a method wherein executing the at least one segment of computer code comprises integrating the at least: one segment of computer code into the computer program and executing the computer program to process the input and obtain the output comprising the type of content that is unknown to the computer program (Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide. When an application runs along with the

original program, Doyle states that it is "embedded" (column 6, lines 58-59, Doyle).

Embedded is equivalent to the claimed integrated).

6.   With regards to claim 30, Doyle teaches a computer system for accessing

multiple types of electronic content, comprising: a processing unit; a memory in

communication with the processing unit; and a computer program stored in the memory

that provides instructions to the processing unit, wherein the processing unit is

responsive to the instructions, operable for: identifying a plurality of segments of

computer code that can be executed along with the computer program by the

processing unit in response to the instructions; selecting, in response to an input

command to access at least one type of content that the computer program is not

configured to access, at least one segment of computer code from the plurality of

segments of computer code that can be executed along with the computer program by

the processing unit, in response to the instructions, to access the at least one type of

content that the computer program is not configured to access; and executing the at

least one segment of computer code along with the computer program to access the at

least one type of content that the computer program is not configured to access,

wherein the plurality of segments of computer code and the at least one segment of

computer code are not executable as an independent computer program (The examiner

concludes after consulting with other members of the office that the phrase "segment of

computer code" is equivalent to programs, program objects and applications. Programs

and applications are segments of computer code. The terms application and program

may be used interchangeably in the disclosure due to their equivalency to the phrase

"segment of computer code." In addition, Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed).

7.   With regards to claim 31, Doyle teaches a computer system wherein the processing unit, responsive to the instructions, is further operable for: arranging in the memory the at least one segment of computer code and a data, comprising the at least one type of content that the computer program is not configured to access, into a function-content group; and interfacing the function-content group to the computer program to enable the computer program to access the at least one type of content that the computer program is not configured to access (Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together, hence two or more may be selected as claimed).

8.   With regards to claim 32, Doyle teaches a computer system wherein the processing unit, responsive to the instructions, is operable for identifying a plurality of segments of computer code by: locating at least two segments of computer code from the plurality of segments of computer code that each comprise a portion of computer code that indicates they can be executed by the processing unit along with the computer program; and generating a list in the memory comprising an identifier for each

of the at least two segments of computer code that indicates that the at least two

segments of computer code are available to be executed by the processing unit along

with the computer program (Doyle teaches a design that features a "list of applications,"

(column 15, line 14, Doyle). The list of applications allows an application to be selected

to run along with the original program to provide an output from an input that the original

program alone would be unable to provide, as claimed. No limitation is given as to how

many applications may be selected to run together; hence two or more may be selected

as claimed).

9.   With regards to claim 33, Doyle teaches a computer system wherein the

processing unit, responsive to the instructions, is operable for selecting at least one

segment of computer code by selecting at least two segments of computer code from

the plurality of segments of computer code whose combined functionality will allow the

computer program to access the at least one type of content that the computer program

is not configured to access when the at least two segments of computer code are

executed by the processing unit along with the computer program (Doyle teaches a

design that features a "list of applications," (column 15, line 14, Doyle). The list of

applications allows an application to be selected to run along with the original program

to provide an output from an input that the original program alone would be unable to

provide, as claimed. No limitation is given as to how many applications may be

selected to run together; hence two or more may be selected as claimed).

10.   With regards to claim 34, Doyle teaches a computer system wherein the

processing unit, responsive to the instructions, is further operable for configuring the at

least two segments of computer code to be executed by the processing unit along with

the computer program in a particular order to allow the computer program to access the

at least one type of content that the computer program is not configured to access

(Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle).

The list of applications allows an application to be selected to run along with the original

program to provide an output from an input that the original program alone would be

unable to provide, as claimed. No limitation is given as to how many applications may

be selected to run together; hence two or more may be selected as claimed. Whenever

multiple applications run together to produce a single output, it is inherent that they will

perform in order as claimed).

11.   With regards to claim 35, Doyle teaches a computer system wherein the

processing unit, responsive to the instructions, is further operable for configuring the at

least two segments of computer code into a master-slave relationship that causes the

execution of one of the at least two segments of computer code to be dependent on the

execution of another of the at least two segments of computer code (Doyle discloses

how applications can be started as child processes (column 15, line 22, Doyle). When a

child process exists, a parent process must exist. This parent-child relationship is

equivalent to a master-slave relationship).

12.   With regards to claim 36, Doyle teaches a computer-readable medium having

computer-executable instructions for accessing multiple types of electronic content,

comprising: logic for creating a list that comprises information about a plurality of

segments of computer code that can be executed along with a computer program; logic

for choosing at least one segment of computer code from the plurality of segments of
computer code, based on the information in the list, that can be executed along with the
computer program to process a type of data that the computer program is not designed
to process; logic to execute the at least one segment of computer code along with the
computer program in response to an input to provide an output of the type of data that
the computer program is not designed to process, wherein the plurality of segments of
computer code and the at least one segment of computer code are not executable as
an independent computer program (The examiner concludes after consulting with other
members of the office that the phrase "segment of computer code" is equivalent to
programs, program objects and applications. Programs and applications are segments
of computer code. The terms application and program may be used interchangeably in
the disclosure due to their equivalency to the phrase "segment of computer code." In
addition, Doyle teaches a design that features a "list of applications," (column 15, line
14, Doyle). The list of applications allows an application to be selected to run along with
the original program to provide an output from an input that the original program alone
would be unable to provide, as claimed).

13.   With regards to claim 37, Doyle teaches a computer-readable medium further
comprising logic for choosing at least two segments of computer code from the plurality
of segments of computer code, based on the information in the list, which can be
executed along with the computer program to process a type of data that the computer
program is not designed to process (Doyle teaches a design that features a "list of
applications," (column 15, line 14, Doyle). The list of applications allows an application

to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together, hence two or more may be selected as claimed).

14.   With regards to claim 38, Doyle teaches a computer-readable medium further comprising logic for linking the at least two segments of computer code in a specific order of execution to provide a desired output of data that the computer program is not designed to process when the at least two segments of computer code are executed along with the computer program (Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together; hence two or more may be selected as claimed. Whenever multiple applications run together to produce a single output, it is inherent that they will perform in order as claimed).

15.   With regards to claim 39, Doyle teaches a computer-readable medium wherein the logic for creating a list that comprises information about a plurality of segments of computer code comprises: logic for identifying at least two segments of computer code that each comprise a registration code that indicates that they can be executed along with the computer program; and logic for generating a list comprising an identification code for each of the at least two segments of computer code that indicates that the at least two segments of computer code are available to be executed along with the

computer program (Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together; hence two or more may be selected as claimed. Whenever multiple applications run together to produce a single output, it is inherent that they will perform in order as claimed. Furthermore, Doyle describes a design that has the means by which to identify the application to be launched from the local user client machine, (column 15, lines 18-21, Doyle)).

16.   With regards to claim 40, Doyle teaches a computer-readable medium further comprising logic for arranging the at least one segment of computer code and a data element, comprising the type of data that the computer program is not designed to process, into a function-data group; and logic for interfacing the function-data group to the computer program to enable the computer program to provide the output of the type of data that the computer program is not designed to process (Doyle teaches a design that features a "list of applications," (column 15, line 14, Doyle). The list of applications allows an application to be selected to run along with the original program to provide an output from an input that the original program alone would be unable to provide, as claimed. No limitation is given as to how many applications may be selected to run together; hence two or more may be selected as claimed. Furthermore, for data to be processed, it typically is grouped together, for instance within a data structure of some form).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 26-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Doyle in view of Tolin et al (US Pat No: US005490061A), hereafter referred to as Tolin.

17.   With regards to claim 26, Doyle teaches through Tolin, a method wherein

receiving a request for a computer program to process an input comprises receiving a

command to translate a word from a first language to a second language

(Doyle discloses a design that features a "list of applications," (column 15, line

14, Doyle). These applications are able to run along with an original program to provide

an output from an input the original program is unable to produce alone. The disclosure

however fails to specify that translation services are offered within that list, to enable the

translating of data from a first language to a second language.

Tolin however discloses a system that performs translation services (column 2,

lines 59-65, Tolin). An application is a system and thus, Tolin's design can be one of

the applications in Doyle's "list of applications."

It therefore would have been obvious to one of ordinary skilled in the art, at the time of the invention to have the translation service taught in Tolin's disclosure with the teachings of Doyle's design to provide a method for running embedded program objects in a computer network environment (column 6, lines 50-52, Doyle)).

18.   With regards to claim 27, Doyle teaches through Tolin, a method wherein receiving a request for a computer program to process an input comprises receiving a command to convert a number from a first number format to a second number format.

(Doyle discloses a design that features a "list of applications," (column 15, line 14, Doyle). These applications are able to run along with an original program to provide an output from an input the original program is unable to produce alone. The disclosure however fails to specify that translation services are offered within that list, to enable the translating of numerical data from a first format to a second format.

Tolin however discloses a system that performs translation services (column 2, lines 59-65, Tolin). The computational process of translation is the same as the computational process of converting numerical formats. Hence, since text can be converted in Tolin's design, the means for converting number formats are also present. An application is a system and thus, Tolin's design can be one of the applications in Doyle's "list of applications."

It therefore would have been obvious to one of ordinary skilled in the art, at the time of the invention to have the translation service taught in Tolin's disclosure with the teachings of Doyle's design to provide a method for running embedded program objects in a computer network environment (column 6, lines 50-52, Doyle)).

19.   With regards to claim 28, Doyle teaches through Tolin, a method wherein receiving a request for a computer program to process an input comprises receiving a command to convert a text object from a first text format to a second text format (Doyle discloses a design that features a "list of applications," (column 15, line 14, Doyle). These applications are able to run along with an original program to provide an output from an input the original program is unable to produce alone. The disclosure however fails to specify that translation services are offered within that list, to enable the translating of text data from a first format to a second format.

Tolin however discloses a system that performs translation services (column 2, lines 59-65, Tolin). The computational process of translation is the same as the computational process of converting numerical formats. Hence, since text can be converted in Tolin's design, the means for converting text formats are also present. An application is a system and thus, Tolin's design can be one of the applications in Doyle's "list of applications."

It therefore would have been obvious to one of ordinary skilled in the art, at the time of the invention to have the translation service taught in Tolin's disclosure with the teachings of Doyle's design to provide a method for running embedded program objects in a computer network environment (column 6, lines 50-52, Doyle)).

20.   With regards to claim 29, Doyle teaches through Tolin, a method wherein receiving a request for a computer program to process an input comprises receiving a command to convert a graphical object from a first graphical format to a second graphical format.

(Doyle discloses a design that features a "list of applications," (column 15, line 14, Doyle). These applications are able to run along with an original program to provide an output from an input the original program is unable to produce alone. The disclosure however fails to specify that translation services are offered within that list, to enable the translating of graphical data from a first format to a second format.

Tolin however discloses a system that performs translation services (column 2, lines 59-65, Tolin). The computational process of translation is the same as the computational process of converting graphical formats. Hence, since text can be converted in Tolin's design, the means for converting graphical formats are also present. An application is a system and thus, Tolin's design can be one of the applications in Doyle's "list of applications."

It therefore would have been obvious to one of ordinary skilled in the art, at the time of the invention to have the translation service taught in Tolin's disclosure with the teachings of Doyle's design to provide a method for running embedded program objects in a computer network environment (column 6, lines 50-52, Doyle)).

Response to Arguments

Applicant's amended claims were carefully evaluated along with the arguments. While the changes have been noted, the examiner feels that they are not fully persuasive. Brief explanations are provided below.

Most notable is the change to the phrase "segment of computer code". The examiner concludes after having consulted with other members of the office that the

phrase "segment of computer code" is equivalent to programs, program objects and applications. Programs, data structures, and applications are some representations of segments of computer code because they are comprised entirely of segments of computer code. The Doyle design allows for what is known as "plug-ins". In plug-ins a set of code is adopted by an original set of code to perform task(s) that the original set of code could not perform alone. This ability is still viewed as being equivalent to the design described by the applicant.

In addition, the Tolin art is used, simply to demonstrate that there does exist methods by which to perform translations. When a computer translates data from one form to another, it does not have to discriminate what type of data it is. Since programs tell computer what tasks to perform, and Tolin's program tells the computer to translate languages, the means are present by which to convert data from one format to another (such as translating). It is believed that the means exist because neither the computer nor the program need to discriminate whether to perform their conversion (or translation) tasks on just languages. As far as the program and computer are concerned, the data is simply converted from one format to another in Tolin's design. Hence, the Tolin design was introduced.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within

TWO MONTHS of the mailing date of this final action and the advisory action is not

mailed until after the end of the THREE-MONTH shortened statutory period, then the

shortened statutory period will expire on the date the advisory action is mailed, and any

extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later

than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Azizul Choudhury whose telephone number is 703-305-

7209. The examiner can normally be reached on M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, David Wiley can be reached on 703-308-5221. The fax phone number for

the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the

Patent Application Information Retrieval (PAIR) system. Status information for

published applications may be obtained from either Private PAIR or Public PAIR.

Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should

you have questions on access to the Private PAIR system, contact the Electronic

Business Center (EBC) at 866-217-9197 (toll-free).

AC

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100